

# Asistencia Visual para la Migración de Procesos

Sergio Martig<sup>1</sup>, Silvia Castro<sup>1</sup>, Mercedes Vitturini<sup>1</sup>, Guillermo Trutner<sup>1</sup>, Carlos Alvez<sup>1</sup>, Sandra Di Luca<sup>1</sup>

Javier Echaiz<sup>2</sup>, Jorge Ardenghi<sup>2</sup>

<sup>1</sup> VyGLab, Laboratorio de Investigación y Desarrollo en Visualización y Computación Gráfica

<sup>2</sup> LISiDi, Laboratorio de Investigación en Sistemas Distribuidos

Dpto. de Ciencias de la Computación

Universidad Nacional del Sur

Avda. Alem 1253

8000 - Bahía Blanca, Argentina

{smartig,smc,mvitturi,cea,je,jra}@cs.uns.edu.ar

{gtrutner,sdiluca}@criba.edu.ar

## Resumen

La migración de procesos con un esquema dinámico involucra una serie de toma de decisiones que deben ser llevadas a cabo en forma rápida y ajustada de modo de no interferir con el aumento de rendimiento y procesamiento total que se busca. Se debe determinar qué tareas, cuándo y hacia dónde deben migrarse, y considerar los costos de las distintas alternativas en cuanto a procesamiento y comunicación. Este planteo de objetivos deriva en un sinnúmero de problemas abiertos; el más relevante es encontrar una métrica que permita tomar las decisiones necesarias para lograr el balance de carga. Para esto resulta de gran importancia contar con una herramienta que permita visualizar el estado de carga de un sistema distribuido. Describimos cuatro prototipos de visualización orientados a obtener una visión del estado de carga del sistema, tanto a nivel de la aplicación distribuida como de cada uno de sus componentes. Delineamos además las etapas cumplidas teniendo como meta no sólo su uso, sino también su usabilidad y presentamos los resultados de las entrevistas con expertos en el dominio.

**Palabras Clave:** Visualización de Información - Computación Gráfica - Sistemas Distribuidos

## 1. Introducción

Un algoritmo de balance de carga trata de equilibrar la carga total de un sistema transfiriendo en forma transparente la carga de trabajo de nodos con cargas pesadas a nodos más descargados, aumentando el paralelismo de ejecución lo que lleva al lógico aumento del rendimiento y del procesamiento total. La principal herramienta que se utiliza para este objetivo es la migración de procesos o datos.

Implementar la migración de procesos involucra una serie de toma de decisiones. Éstas deben ser realizadas de modo de minimizar la interferencia en el aumento de rendimiento y procesamiento total. El algoritmo de migración debe decidir qué proceso mover, hacia dónde moverlo y en qué momento debe hacerse. A su vez, estas acciones deben demorar el menor tiempo posible. Este planteo de objetivos deriva en un sinnúmero de problemas abiertos o que tienen soluciones parciales adecuadas a la aplicación. El problema más relevante lo constituye encontrar una métrica posible que permita tomar las decisiones necesarias para lograr el balance de carga.

Este trabajo surge entonces como una necesidad de contar con una herramienta que permita visualizar el estado de carga de un sistema distribuido. Fueron desarrollados cuatro prototipos que permiten visualizar determinados parámetros de un sistema distribuido, de manera que a golpe de vista sea posible poder obtener una idea del comportamiento del estado de carga del sistema, tanto a nivel de la aplicación distribuida como de cada uno de sus componentes.

A partir de esta visualización, el usuario experto podrá tomar las acciones que considere adecuadas para sintonizar el sistema o modificar el algoritmo. Si considera conveniente que para estabilizar la carga del sistema, por ejemplo, debe migrar procesos, la visualización lo asistirá en dicha toma de decisión. De este modo, el contar con una herramienta gráfica que permita visualizar el estado del sistema distribuido, ayudará a tomar decisiones y permitirá una mejor y más rápida comparación del estado de los distintos nodos involucrados. Este trabajo surge en el marco del Dpto. de Cs. de la Computación, Universidad Nacional del Sur, Bahía Blanca como un trabajo interdisciplinario entre los laboratorios de Visualización y Computación Gráfica (VyGLab) y del Laboratorio de Investigación en Sistemas Distribuidos (LISiDi).

En la próxima sección se presenta la formalización del problema del balance de carga, al que se asiste en su solución mediante el uso de las herramientas visuales diseñadas, y se establecen las restricciones asumidas. Luego se describen cada uno de los prototipos y a continuación se presentan los resultados logrados en las distintas etapas del diseño. Finalmente se presentan las conclusiones obtenidas y los desarrollos en curso.

## 2. Especificación del problema

El balance de carga constituye una herramienta teórica muy adecuada para lograr mayor rendimiento y procesamiento total del sistema, pero en la práctica plantea problemas que no son inmediatamente solubles debido a todas las restricciones enumeradas. Esto plantea la necesidad de contar con una herramienta que permita representar gráficamente el estado de carga de un sistema distribuido. A simple vista debe ser posible poder obtener una idea del estado de carga del sistema, tanto a nivel de la aplicación distribuida como de cada uno de sus componentes.

Los items más importantes a considerar cuando se pretende compartir la carga son:

- *Determinar hacia dónde enviar una tarea.* Las opciones más comunes involucran prueba, multicasting o centralizado. El método de prueba consiste en enviar una solicitud a varias máquinas para determinar si está disponible alguna con carga baja. Los métodos de mantener información centralizada de la carga y multicasting suelen ser costosos y no siempre escalan bien. La decisión en este caso fue la elección de un método de prueba guiado por la información que se tiene sobre el estado actual del sistema.
- *Definir qué se entiende por carga alta y qué por carga liviana.* Esto está fuertemente ligado con determinar cuáles son los parámetros del sistema que deben monitorearse para poder determinar el nivel de carga. Algunos ejemplos consideran la cantidad de procesos esperando por la CPU, el porcentaje ocioso de la máquina, etc. En este trabajo las métricas que se consideraron de interés para poder estimar la carga de cada una de las máquinas comprenden la capacidad de memoria de la máquina y la memoria disponible, el porcentaje de utilización de la CPU, la cantidad de procesos corriendo en cada una de las estaciones y la cantidad y volumen de los mensajes entrantes y salientes.

Una característica que tiene este problema es que en principio no se puede predecir cuándo el sistema se encontrará no balanceado. La visualización estará siempre disponible para que los usuarios, administradores del sistema en este caso, tomen la información que necesiten en cualquier momento. Ante una situación de inestabilidad, el sistema no será el responsable de la toma de decisión. Todos los prototipos que se presentan en este trabajo no ofrecen características que permitan predecir situaciones que requieran la necesidad de tomar la decisión, ni tomarán decisiones, ni postularán alternativas de solución. Este rol será totalmente asumido por los usuarios expertos en el tema.

El sistema que se quiere visualizar está conformado por  $N$  nodos unidos por una conexión punto a punto. Cada nodo está caracterizado por un número que lo identifica, un tamaño de memoria y los porcentajes de CPU ocupada. En cada nodo se ejecutan hasta  $M$  procesos ( $M > 1$ ). La información de cada proceso corresponde a la memoria que usa el proceso, la cantidad de archivos abiertos, la

cantidad de mensajes entrantes y salientes que tiene el proceso y el tamaño promedio de los mismos.

Estos son los parámetros que se establecieron para visualizar con el objetivo de poder determinar cuáles son las máquinas con carga alta y cuáles las más ociosas. Se asume que todos estos datos se informan a la visualización. En principio, no se visualizará información sobre la topología de la red, ni sobre cuáles son los procesos que intercambian mensajes.

Se asume que de acuerdo a lo planteado, el sistema de visualización no tomará decisiones propias, ni propondrá alternativas; serán los usuarios quienes asumirán este rol a partir de la información que surja de la visualización. El sistema se limitará a ofrecer ayuda visual para la toma de decisión. Cuando los usuarios así lo requieran podrán, a partir de la visualización, identificar los problemas y, si lo consideran oportuno, ejecutar las acciones pertinentes. Actualmente, el sistema de visualización no está trabajando en tiempo real.

Se están estudiando las ventajas y desventajas de cada una de las aplicaciones y la calidad de la información ofrecida para poder identificar rápidamente cuáles son los nodos sobrecargados, cuáles son los procesos candidatos a migrar, y las posibles máquinas a las que se migrarán.

### **3. Distintos prototipos de visualización**

Se desarrollaron cuatro prototipos diferentes que visualizan la misma información básica pedida para un mismo conjunto de datos de entrada; las cuatro alternativas presentadas se desarrollaron en MatLab. Los indicadores de carga del sistema distribuido sugeridos para ser visualizados podemos dividirlos en:

1. Parámetros generales del nodo:
  - Tamaño de la memoria
  - Porcentaje de ocupación de CPU
2. Parámetros a nivel de cada uno de los procesos en un nodo
  - Tamaño en memoria de cada proceso
  - Cantidad de archivos abiertos
  - Cantidad de mensajes enviados
  - Volumen de la información enviada
  - Cantidad de mensajes recibidos
  - Volumen de la información recibida

Los diferentes prototipos que se presentan en este trabajo son el resultado de una constante interacción con los usuarios del sistema en la evolución hacia posibles soluciones. En el ciclo de desarrollo, algunos de los prototipos fueron desechados o modificados. Todas las aplicaciones fueron testeadas con datos concretos, provistas por los usuarios. Las primeras soluciones encontradas apuntaron hacia visualizaciones centradas en el nodo. En la medida en que el número de parámetros a visualizar fue creciendo, muchas debieron ser descartadas ya que no era posible mostrar tanta información en un esquema que represente a cada nodo. Finalmente, sólo una de las alternativas pudo seguir siendo centrada en el nodo porque incorporó, fundamentalmente, una tercera dimensión que le permitió poder mostrar todos los parámetros. A continuación se analizarán distintos aspectos presentados por cada una de las diferentes alternativas que surgieron para visualizar el balance de la carga.

### 3.1. Representación de los nodos y sus atributos

La visualización de los nodos deben dar al usuario experto en Sistemas Distribuidos una primera aproximación instantánea del estado de carga del sistema. Por ello, en cada uno de los prototipos desarrollados se buscó transmitir esta situación a golpe de vista. La información que necesariamente debe presentarse en cada nodo es el tamaño de la memoria y el porcentaje de ocupación de CPU. Debido al tipo de asistencia que se debe brindar resulta fundamental mostrar el tamaño de la memoria de cada nodo y la disponibilidad de ese recurso. En un sistema con componentes no homogéneas es primordial que los tamaños de las memorias de los distintos nodos sean mostrados de una manera efectiva para posibilitar su comparación.

#### *Prototipo 1: Visualización 3D*

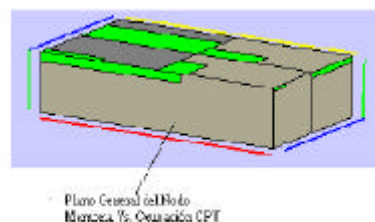
Esta es una visualización centrada en los nodos. Los indicadores mostrados se agruparon consecuentemente por nodos, utilizando el color y la disposición de los gráficos para reforzar la comunicación de esa información y facilitar la comparación de los distintos indicadores para los nodos analizados. Para poder lograr una uniformidad en la visualización de todos los indicadores relacionados con un nodo, se decidió mapearlos sobre las tres caras visibles de un prisma rectangular dibujado en 3D.

En la Figura 1 se muestra la visualización para un nodo con dos procesos. Un prisma rectangular representa tanto al nodo como a los atributos referidas a los procesos. La información general correspondiente al nodo se representa en una de las caras del prisma y la denominaremos Plano General del Nodo. En éste se mapean:

- Tamaño de la memoria del Nodo:

Está representada por la altura total de la cara que varía proporcionalmente en cada nodo, en función del nodo con mayor memoria.

- Porcentaje de ocupación de CPU: La base de la cara representa el valor de referencia para la ocupación de CPU (100%)



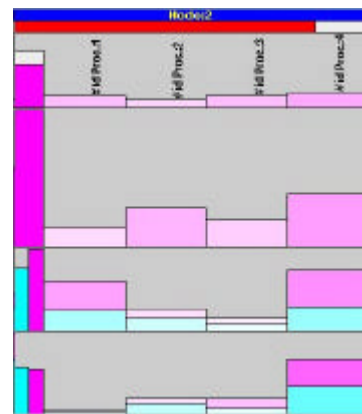
**Figura 1. Representación 3D del Nodo.**

El rectángulo coloreado dibujado a partir del ángulo superior izquierdo de la cara representa la cantidad total de memoria ocupada del nodo versus el porcentaje de ocupación de la CPU. Así se tiene una visión rápida y clara de los indicadores generales del nodo favoreciendo, por su forma y escala utilizada, la comparación de los mismos.

#### *Prototipo 2: Visualización orientada a los nodos*

Este prototipo permite visualizar varios atributos de un sistema distribuido para facilitar la comparación entre nodos. Para este propósito los datos se muestran organizados en filas y columnas, donde cada columna representa un nodo y cada fila un atributo.

Cada columna identifica un nodo. En la fila superior se muestra el porcentaje de uso de CPU en una barra horizontal; en la primera barra de la fila siguiente se muestra la memoria ocupada y libre de cada nodo y en la tercera fila se muestra la cantidad de archivos abiertos en el nodo en la primer. Los mensajes tanto en cantidad como en volumen se muestran en la cuarta y quinta fila respectivamente, en



**Figura 2. Visualización del nodo en el prototipo 2.**

ambos casos, las primeras dos barras se utilizan para representar el total de mensajes entrantes (cyan) y salientes (magenta) respectivamente.

#### *Prototipo 3: Visualización con barras*

La visualización obtenida a partir de los datos recibidos es la presentada en la Figura 7. La pantalla está organizada en dos sectores bien diferenciados. En el sector izquierdo se muestra información acerca de los parámetros de los nodos, mientras que en el lado derecho se muestra la información correspondiente a los parámetros de los procesos.

Los cinco gráficos de barras ubicados en el sector de los nodos (Figura 3) corresponden respectivamente al tamaño de los nodos, al porcentaje de ocupación de CPU, a la cantidad de archivos abiertos, a la cantidad de mensajes entrantes y salientes y al tamaño de los mensajes entrantes y salientes. En cada gráfico, cada barra corresponde a un nodo.

En el caso de la cantidad de archivos abiertos, la cantidad de mensajes entrantes y salientes y el volumen de mensajes entrantes y salientes se muestra la suma de los indicadores correspondientes a los distintos procesos en cada nodo. En el caso de los mensajes entrantes y salientes, del eje central hacia la derecha se indican los mensajes entrantes mientras que hacia la izquierda son los salientes.

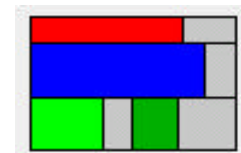


**Figura 3. Representación de los nodos en Prototipo 3**

#### *Prototipo 4: Visualización híbrida con coordenadas paralelas*

Cada nodo se representa mediante un ícono y cada proceso mediante un punto en coordenadas paralelas. Así se tiene, a primera vista, una idea de lo que está ocurriendo en cada uno de los nodos del sistema.

El ícono del nodo se divide en tres franjas: la inferior representa los mensajes, la intermedia, el porcentaje de ocupación de CPU y la superior, la cantidad de memoria ocupada del nodo; en tanto las franjas inferior y central tiene una altura constante, la altura de la superior varía de acuerdo a la cantidad de memoria del nodo. A la mayor memoria de todos los nodos se le asigna una determinada altura y la altura de los rectángulos de los demás nodos, se ajusta relativa a ésta. En sentido horizontal se presenta la cantidad de memoria ocupada de dicho nodo .



**Figura 4. Ícono del Nodo**

La franja inferior corresponde a las características de los mensajes entrantes (izquierda) y salientes (derecha). En el eje horizontal se representa cantidad de mensajes, en tanto que en el vertical se representa el volumen de los mismos. Estas cantidades también representan valores relativos a la mayor cantidad de mensajes y mayor volumen de información, en un determinado instante. Así se tiene, a primera vista, una idea de lo que está ocurriendo en cada uno de los nodos del sistema.

### **3.2. Representación de los procesos y sus atributos**

La información a visualizar es relativa a los nodos. Cada nodo puede tener varios procesos, siendo el proceso la granularidad de migración.

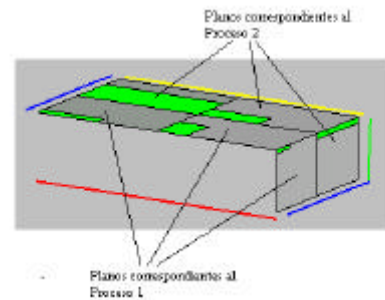
#### *Prototipo 1*

Los procesos se representan en las otras dos caras visibles del prisma rectangular. Cada proceso se representa sobre cada plano en forma consecutiva mediante rectángulos cuyas dimensiones (iguales para todos los procesos) surgen del análisis de los indicadores correspondientes a todos los procesos

en todos los nodos. Así se favorece la comparación entre los distintos indicadores entre todos los procesos independientemente del nodo al que pertenezcan.

- Plano Cantidad de Archivos Abiertos versus Tamaño en Memoria del proceso:

- La altura de cada rectángulo en esta cara representa el total de memoria disponible en el nodo.
- La base de los rectángulos en esta cara representa el valor de referencia para la cantidad de archivos abiertos, correspondiente a la cantidad máxima de archivos abiertos por un proceso en algún nodo.



**Figura 5. Representación 3D de los Procesos.**

- Plano Cantidad de Mensajes Entrantes/Salientes versus Volumen de Información Recibida/Enviada:

- La altura de cada rectángulo se corresponde con la máxima cantidad de mensajes Entrantes/Salientes entre todos los procesos de todos los nodos.
- La base de cada rectángulo se corresponde con el máximo volumen de información Recibida/Enviada por todos los procesos de todos los nodos.

El rectángulo coloreado dibujado a partir del ángulo superior izquierdo del rectángulo asociado a cada proceso en los distintos planos representa los valores correspondientes al proceso en cuestión.

#### *Prototipo 2*

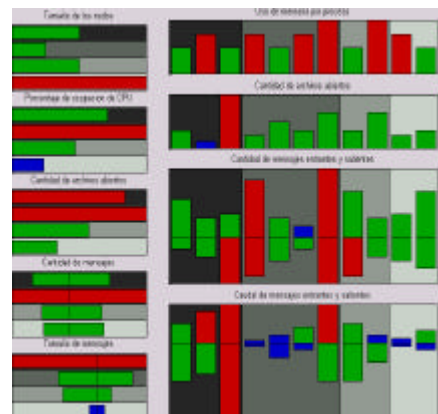
Los valores de los atributos de los procesos se muestran en las filas respectivas dentro de las columnas asociadas con los nodos a los cuales pertenecen. La memoria utilizada y los archivos abiertos se representan directamente mediante barras cuya altura indica el valor de los mismos. Los mensajes entrantes y salientes se muestran en forma de barras apiladas, mostrando los salientes por encima de los entrantes.

#### *Prototipo 3*

Los cuatro gráficos de barras en el sector de los procesos (Figura 6), representan uso de la memoria por proceso, cantidad de archivos abiertos, cantidad de mensajes entrantes y salientes, y volumen de mensajes entrantes y salientes respectivamente. En cada gráfico, cada barra representa un proceso, indicando el fondo la pertenencia al nodo correspondiente.

En el caso de mensajes entrantes y salientes, los mensajes entrantes están del eje central hacia arriba y los mensajes salientes están del eje central hacia abajo.

En ambas representaciones, cada gráfico se escala a partir del mayor de los valores para dicho parámetro, y los demás valores se escalan a partir del mismo.

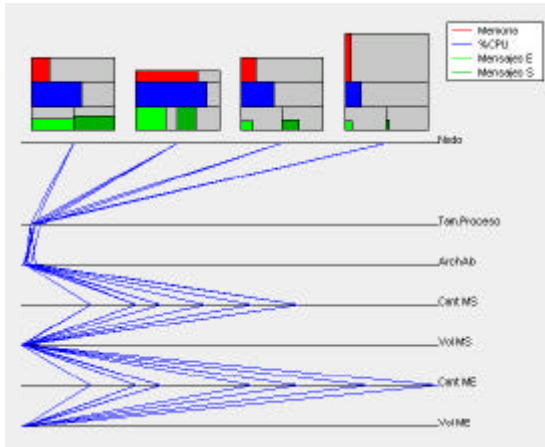


**Figura 6. Visualización de los procesos**

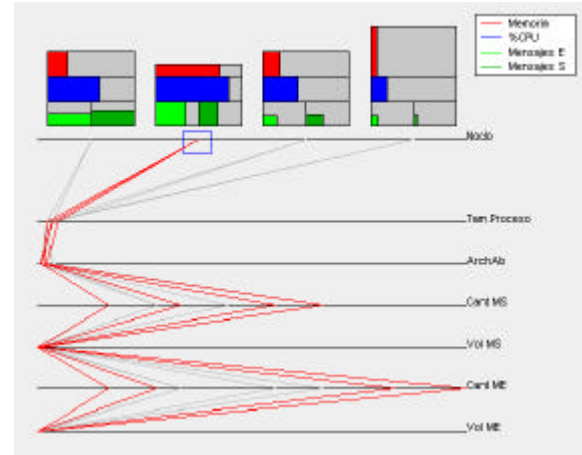
#### *Prototipo 4*

La información relativa a los procesos se grafica en coordenadas paralelas; cada proceso representa un punto en dicho sistema de coordenadas. Así, cada proceso se grafica mediante una línea que parte del nodo respectivo y corta a cada uno de los ejes en el valor correspondiente a un atributo. Los atributos de los procesos se grafican en valores absolutos. De arriba hacia abajo se representa el tamaño del proceso, la cantidad de archivos

abiertos, la cantidad de mensajes de salida, el volumen de los mismos, la cantidad de mensajes de entrada y el volumen de estos (Figura 7).



**Figura 7. Alternativa 4 para un sistema con 4 nodos**



**Figura 8. Interacción restringiendo sobre un Nodo**

Además, se permite seleccionar un rango cualquiera de un determinado atributo. En la Figura 8 se muestran activados (en rojo), todos los procesos cuya cantidad de mensajes de entrada está dentro del rango señalado por la caja azul.

### 3.3. Uso del color

#### *Prototipo 1*

En esta visualización se ha utilizado el color con dos propósitos distintos:

- *Transmitir en forma redundante el porcentaje de utilización de CPU de cada nodo.* Con este fin se utiliza una asociación cromática con tres estados; a los estados de alta ocupación, ocupación normal y baja ocupación se les asocia respectivamente rojo, verde y azul. Una vez determinado el estado de uso de CPU y con el propósito de proveer información del estado del nodo a primera vista, se utiliza el mismo color correspondiente para todas las áreas coloreadas del mismo nodo.
- *Unificar los planos de referencia.* Dado que los indicadores para cada nodo se visualizan agrupados, el color de fondo se eligió de modo tal que facilite la interpretación y posterior comparación entre los distintos parámetros mostrados para todos los nodos. Con este fin se utiliza el mismo color de fondo para cada uno de los planos de referencia en los distintos nodos.

En particular, se trabaja con una escala de grises por considerarse que no compite, sino que realza, los colores relevantes para la información mostrada.



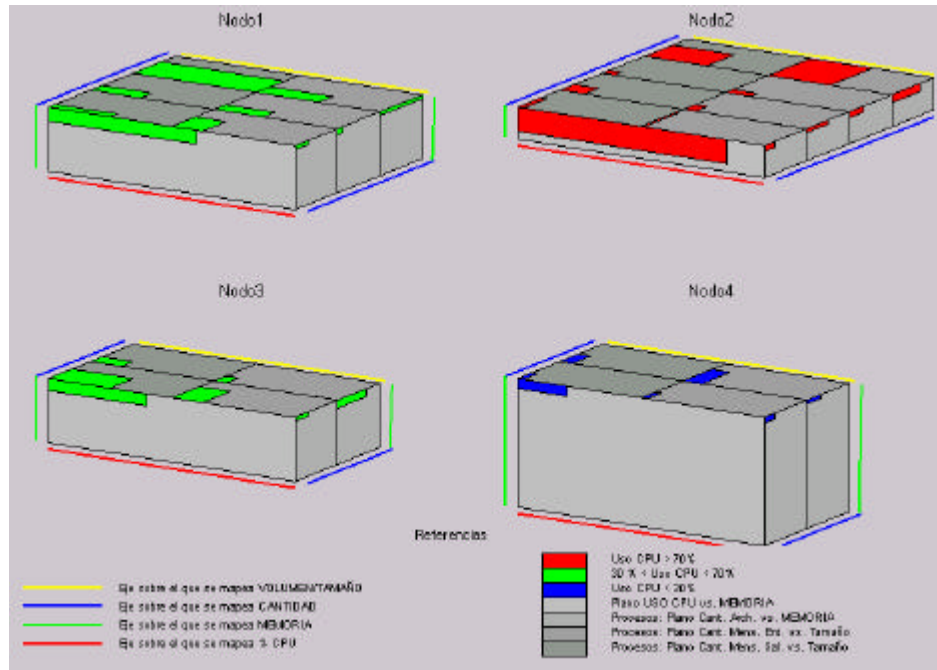


Figure 9. Visualización 3D del sistema

## Prototipo 2

Dentro de cada nodo, la saturación del color muestra en forma redundante los valores de los atributos.

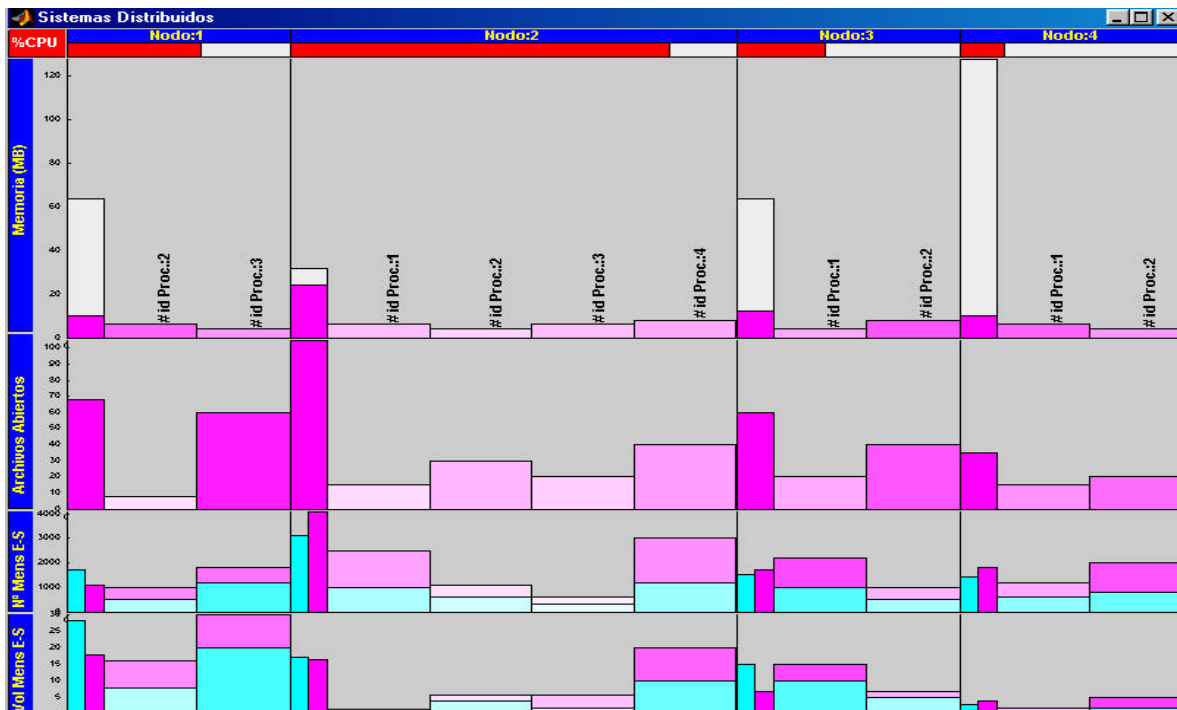


Figure 10. Visualización del sistema de acuerdo al Prototipo 2. Datos organizados en filas y columnas



### Prototipo 3

En esta visualización también se utiliza una asociación cromática de tres estados relativa al peso de cada parámetro referido al máximo valor; así, rojo indica valores por encima del 75% del máximo de referencia, verde entre el 25% y el 75% y azul menos del 25%. Las distintas intensidades de grises utilizadas como fondo establecen la correspondencia de parámetros con los respectivos nodos asociados.

Esta interface permite una interacción que puede ser útil en caso de querer individualizar la información de un nodo. Seleccionando cualquier zona de la pantalla que pertenezca a un cierto nodo, todos las zonas de la pantalla pertenecientes a dicho nodo quedan iluminadas mientras que el resto se opaca (Figura 11).

### Prototipo 4

Tanto en los íconos como en las coordenadas paralelas, los colores han sido elegidos ortogonales, de modo tal que permitan distinguir claramente las características que se quieran representar. Cuando, como en el caso de la Figura 8, se seleccionan determinados datos, estos se resaltan en color rojo en tanto que los no seleccionados se detallan en gris; así se logra que no se pierda el contexto pero que se resalte la información de interés.

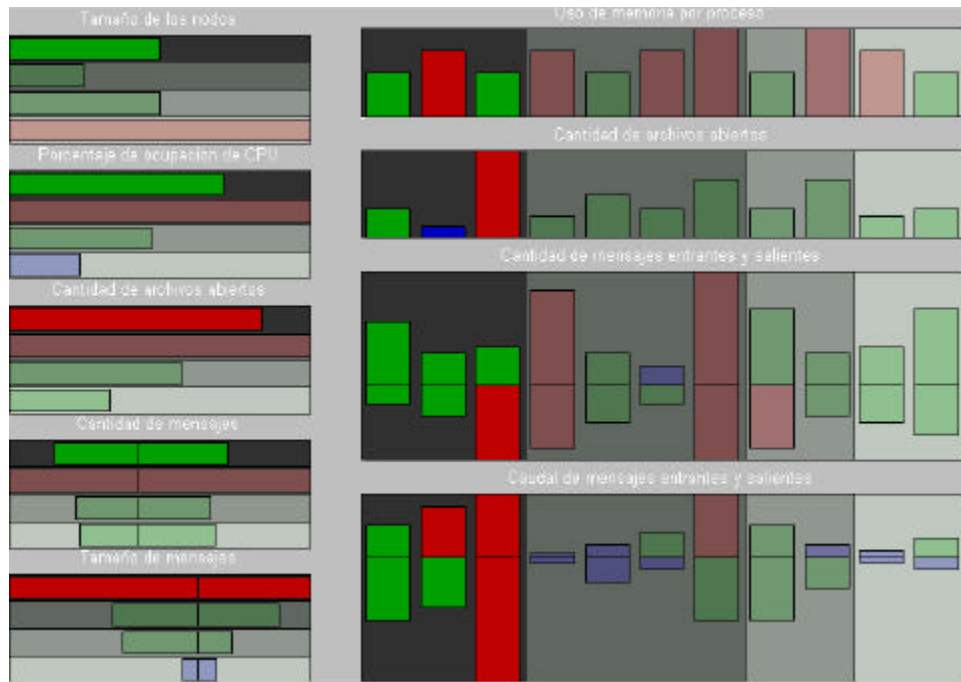


Figura 11. Ejemplo de interacción para la visualización de un nodo

## 4. Diseño de la visualización centrado en el usuario

Nuestro objetivo fue diseñar un sistema flexible que permitiera transmitir de manera efectiva el estado de carga de un sistema distribuido. Nos encaminamos hacia esta meta utilizando un proceso de diseño centrado en el usuario.

Para realizar este sistema utilizamos un modelo de desarrollo basado en prototipos con interacción constante con los usuarios de la visualización, buscando refinar el sistema y sus componentes y concentrándonos desde el comienzo del proyecto en la usabilidad del producto.

El estudio del problema y el relevamiento de los requerimientos funcionales y de usabilidad de una herramienta de estas características sirvió de punto de partida. Para esto realizamos una serie de entrevistas semiestructuradas con expertos en el área de Sistemas Distribuidos para explorar el

potencial uso de la visualización. Fueron entrevistados, en varias oportunidades, cinco expertos en el dominio quienes en principio serán los usuarios finales de la visualización. Para controlar el avance del proyecto lo organizamos en tres etapas principales. Cada una de las etapas a su vez las dividimos en entrevistas a los usuarios y repercusión de los resultados en el diseño.

#### **4.1 Entrevistas**

Todas las entrevistas realizadas a los usuarios las desarrollamos siguiendo una guía de preguntas previamente elaborada para poder conducir la conversación hacia los objetivos preestablecidos.

*Primera etapa: Presentación del problema y planteo de requerimientos.*

Antes de contar con los prototipos entrevistamos a los usuarios para obtener una presentación general del problema y especificar qué aspectos querían visualizar. Indagamos sobre cuáles eran las herramientas que estaban utilizando y por qué les resultaban inconvenientes y/o incompletas. Además y debido a que las herramientas a construir debían ayudar en el proyecto de investigación sobre el balance de carga, hicimos especial énfasis en definir cuáles serían los parámetros del sistema a visualizar.

*Segunda etapa: Evolución de los prototipos*

Mostramos a los usuarios los prototipos en todas sus versiones. Las entrevistas tuvieron como objetivo saber si cada una de las visualizaciones respondía a las expectativas planteadas. Para las versiones finales, se amplió el espectro de usuarios con otros usuarios potenciales no involucrados en el proyecto.

*Tercera etapa: Especificación de interacciones*

A esta altura, las aplicaciones estaban definidas desde el punto de vista de la visualización. Nos concentramos entonces en discutir con los usuarios las interacciones que se podrían agregar a cada uno de los prototipos de manera de aumentar su usabilidad.

#### **4.2 Resultados Obtenidos**

A continuación mostramos cómo los resultados obtenidos en las entrevistas impactaron en el diseño de las aplicaciones.

*Primera etapa*

Como resultado de las entrevistas concluimos que todos los expertos coincidieron en que un factor importante para la visualización era poder tener, a golpe de vista, una idea general del estado del sistema a nivel de nodo. Sin embargo no pasó lo mismo con la definición de los parámetros a visualizar; en este caso, si bien no hubo total coincidencia en cuanto a los parámetros generales a visualizar de un determinado nodo, sí se logró consenso en que era fundamental mostrar el porcentaje de CPU usado y la cantidad de memoria utilizada y disponible de cada nodo. En cuanto a la definición de los parámetros a visualizar para cada uno de los procesos, finalmente definimos los que quedaron detallados en la sección 3 de este trabajo, aunque no hubo acuerdo en si deberían mostrarse cantidades absolutas o relativas.

*Segunda etapa*

En función a los datos recabados en las entrevistas, desarrollamos los cuatro prototipos que presentamos en este trabajo. Los mismos fueron expuestos y depurados por los expertos quienes los analizaron críticamente. Como resultado concluimos que:

- En general las visualizaciones fueron adecuadas notándose entre ellas características distintivas, pero importantes en cada una.
- Las distintas posibilidades que se brindaban en cuanto a enfocar sobre valores relativos o absolutos, o sobre rangos específicos de esos valores, fueron citadas como ventaja ya que, debido a la naturaleza experimental de la investigación que deben realizar, se presentan como alternativas de gran utilidad.

- Los códigos de color utilizados fueron claros e intuitivos en todos los casos y surgió la alternativa, en algunos de ellos de poder contar con un código que identificara distintos rangos de valores relevantes de los parámetros (por ejemplo, menor que el normal, normal, mayor que el normal).
- Se consideró muy efectivo el uso de prototipos iniciales para guiar la discusión y generar así las distintas alternativas que soporten las distintas decisiones de diseño.
- El contar con los prototipos y ver que podían ayudar en gran medida para lograr avances en la investigación planteada, hizo que surgieran ideas en cuanto a otras alternativas a visualizar tales como una caracterización de los procesos en cuanto a si son procesos de CPU o *I/O bounded*, información acerca de los procesos hijos o sea condición de *arrastre* y, como información estática adicional, la velocidad de comunicación de la CPU y los canales internos.

### *Tercera Etapa*

El permitir a los usuarios interactuar con los prototipos condujo al planteo de contar con otras formas de interacción, además de las presentadas en las distintas alternativas. Es por esto que actualmente estamos trabajando en una segunda generación de prototipos que utilizaremos para refinar progresivamente la interacción. Sin duda, las técnicas de interacción deben facilitar el proceso de exploración de los datos.

Si bien inicialmente algunas de las propuestas resultaron más simples de leer que otras, se concluyó que la información que brindan resulta altamente relevante tanto para el estudio experimental del problema de balance de carga, como para la toma de decisión en problemas de estas características

## **5. Conclusiones**

Se presentaron cuatro prototipos para visualizar el estado de carga de un sistema distribuido para asistir en la toma de decisiones sobre qué procesos migrar y hacia qué máquinas migrarlos, con el objetivo de aumentar el rendimiento total del sistema. También se mostraron los pasos que guiaron el desarrollo de los mismos siguiendo una metodología centrada en el usuario. Si bien inicialmente algunas resultaron más simples de interpretar que otras, se concluyó que la información que brindan es altamente relevante tanto para la toma de decisión en problemas con las características planteadas como en el estudio del problema de balance de carga.

Desde el punto de vista de la visualización, estamos trabajando en incorporar las alternativas presentadas en un sistema único de visualización ya que sin interfaces efectivas, el usuario no puede utilizar adecuadamente las técnicas subyacentes para extraer información.

Desde el punto de vista de la interactividad, inicialmente permitirá la selección de distintos parámetros y de distintos subconjuntos de valores de los mismos de acuerdo a lo que resultó de las entrevistas de los expertos con las distintas visualizaciones. También se examinarán otras técnicas de interacción que faciliten el proceso de exploración de la visualización.

Desde el punto de vista de los investigadores en Sistemas Distribuidos, esta herramienta será utilizada con el objeto de obtener algún tipo de correlación entre las distintas métricas propuestas de modo de lograr una parametrización ajustada al objetivo del balance de carga.

## **Bibliografía**

- 1 Card, S., Mackinlay, J., Shneiderman, B., *Readings in Information Visualization – Using Vision to Think*, Morgan Kaufmann, 1999.
- 2 Coulouris, G., Dollimore, J. y Kindberg, T., *Distributed Systems, Concepts and Design*, Addison-Wesley, 3<sup>rd</sup>. Edition, 2001.

- 3 Foley, J., Van Dam, A., *Fundamentals of Interactive Computers Graphics*, Addison-Wesley, Reading, Massachussetts, segunda edición, 1992.
- 4 Gallagher, R., *Computer Visualization: Graphics Techniques for Sc. And Eng. Analysis*, 1996.
- 5 Inselberg, A., and Dimsdale, B., *Multidimensional lines: Representation*, SIAM J. Appl. Math., Vol. 54, No. 2, 1994, pp. 559-577.
- 6 Johnson, B., Shneiderman, B., *Treemaps: A Space-Filling Approach to the Visualization of Hierarchical Information Structures*, Proceedings of IEEE Information Visualization '91, pp. 275-282.
- 7 Kelly, P., Keller, M., *Visual Cues: Practical Data Visualization*, IEEE Computer Society Press, 1992.
- 8 Levkowitz, H., Herman, G., *Color Scales for Image Data*, IEEE Computer Graphics and Applications, 12, pp. 78-80.
- 9 Levkowitz, H., Holub, R., Meyer, G., Robertson, A., *Color vs. Black and White in Visualization*, IEEE Computer Graphics and Applications, 12(14), pp. 20-22.
- 10 Nielson, G., Hagen, H., Müller, H. *Scientific Visualization: Overviews, Methodologies and Techniques*, IEEE Computer Society, 1997.
- 11 Robertson, G., Card, S., Mackinlay, J., *Information Visualization Using 3D Interactive Animation*, Communications of the ACM, 36(4), pp. 56-71, 1993.
- 12 Tanenbaum, A., *Sistemas Operativos Distribuidos*, Prentice Hall, 1996.
- 13 Tufte, E.R., *The Visual Display of Quantitative Information*, Cheshire, CT Graphics Press, 1983.
- 14 Tufte, E.R., *Envisioning Information*, Cheshire, CT Graphics Press, 1990.
- 15 Tufte, E.R., *Visual Explanations: Images and Quantities, Evidence and Narrative*, Cheshire, CT Graphics Press, 1997.
- 16 Watt, A., Watt, M., *Advanced Animation and Rendering Techniques: Theory and Practice*, Addison-Wesley Publishing Company, 1993.
- 17 Wong P.C., Bergeron R. D., *30 Years of Multidimensional Multivariate Visualization*, Proc. Workshop on Scientific Visualization, IEEE Computer Society Press, 1995.